Description

Two-Layer Display-Refresh and Video-Overlay Arbitration of Both DRAM and SRAM Memories

BACKGROUND OF INVENTION

[0001]   This invention relates to graphics systems, and more particularly to arbitration of multiple requestors to multiple memory devices.

[0002]   Improvements in semiconductor processing has allowed for larger systems to be integrated together on smaller integrated circuit chips. More powerful graphics engines such as for 3-D rendering and manipulation can be integrated together with basic screen refresh controllers. Advanced functions such as for video-overlay can be integrated with screen refresh controllers.

[0003]   Sometimes video overlay engines and screen refresh controllers access the same physical memory device, such as a graphics dynamic-random-access memory (DRAM). However, higher-resolution, high-color-depth, and high-

speed graphics displays may require the use of faster static random-access memory (SRAM). For example, the frame buffer of pixels to display on the screen during each refresh can be located in a fast SRAM while video objects and textures are stored in a slower DRAM.

[0004] DRAM usually stores data as charges on capacitors that periodically require refreshing of the charges, while SRAM stores data as states of a bi-stable circuit such as a bi-stable latch. The access time for the SRAM is often much smaller than the access time for the DRAM.

[0005] Figure 1 shows a graphics system memory that uses both SRAM and DRAM. SRAM 12 is faster than DRAM 10, so frame buffer 14 is stored primarily in SRAM 12 to improve refresh speed. However, larger screens and pixel sizes may require the use of extension 18 in DRAM 10. Extensions may be needed when frame buffer 14 is larger than the available space in SRAM 12. The frame buffer may have different sizes, depending on whether the display is a cathode-ray tube (CRT) or liquid crystal display (LCD). Some display modes may display two or more display devices, such as when a laptop drives both its LCD and an external CRT or TV monitor.

[0006] More realistic-looking images may be constructed from

3-D objects that are manipulated in a variety of ways, such as by rotation, transformation, shading, blending, transparency, and texturing. A portion of the screen may contain a window displaying a video from a feed or other source different from the rest of the screen. Video overlay processors can perform these advanced video.

[0007] Video overlay engines may require a number of buffers and storage areas in memory. Some buffer areas may store objects in a 3-Dimensional space that are only occasionally accessed. These objects may be stored as video overlay data 19 in slower DRAM 10. Other buffers may be more frequently accessed, such as temporary buffers or video-feed buffers. Video overlay data 16 in SRAM 12 may contain these higher-speed buffers. Thus refresh and overlay data may each be present in both SRAM 12 and DRAM 10.

[0008] What is desired is a graphics system that allows a refresh controller and an overlay engine to access both DRAM and SRAM devices. A bus architecture and arbitration scheme is desired for such as multi-master, multi-memory graphics system.

BRIEF DESCRIPTION OF DRAWINGS

[0009] Figure 1 shows a graphics system memory that uses both

SRAM and DRAM.

[0010] Figure 2 is a block diagram of a simple multi-master, multi-memory-device graphics system.

[0011] Figure 3 shows a single arbiter controlling access to separate memory devices in a 2-layer bus architecture.

[0012] Figure 4 shows a dual-layer arbiter with 3 requestors.

[0013] Figure 5 details signals to and from the dual-layer arbiter with three requestors.

[0014] Figure 6 shows a more sophisticated embodiment of a dual-layer arbiter that prioritizes the refresh controller.

[0015] Figure 7 is a waveform illustrating arbitration using the dual-layer arbiter.

DETAILED DESCRIPTION

[0016] The present invention relates to an improvement in graphics systems. The following description is presented to enable one of ordinary skill in the art to make and use the invention as provided in the context of a particular application and its requirements. Various modifications to the preferred embodiment will be apparent to those with skill in the art, and the general principles defined herein may be applied to other embodiments. Therefore, the present invention is not intended to be limited to the par-

ticular embodiments shown and described, but is to be accorded the widest scope consistent with the principles and novel features herein disclosed.

[0017] Figure 2 is a block diagram of a simple multi-master, multi-memory-device graphics system. Liquid crystal display (LCD) refresh controller 20 writes a stream of pixels to one or more display devices such as a flat-panel LCD screen or a CRT monitor. These pixels are read from a frame buffer that usually resides in SRAM 12, but may be partially in DRAM 10.

[0018] Video overlay engine 22 performs complex graphics functions, such as 3-D rendering and manipulation, or video-feed processing. Overlay data is often in DRAM 10, but may also be located in SRAM 12.

[0019] Arbiter 24 arbitrates requests from refresh controller 20 and from overlay engine 22 for access to SRAM 12. When refresh controller 20 accesses SRAM 12, overlay engine 22 must wait since it generally has lower priority. Likewise, arbiter 26 arbitrates requests from refresh controller 20 and from overlay engine 22 for access to DRAM 10. Again, refresh controller 20 is often given higher access privilege, but since the frame buffer is often not in DRAM 10, overlay engine 22 can often access DRAM 10 without de-

lays.

[0020] Having two separate buses to DRAM 10 and to SRAM 12 allows for concurrent memory access, where one master can access the DRAM while the other master is accessing the SRAM. Since the LCD frame buffer is often in SRAM, or mostly in SRAM, while the video overlay data is mostly in DRAM, refresh controller 20 can access SRAM 12 while overlay engine 22 is accessing DRAM 10. On the occasions when both masters desire to access the same memory, "real" arbitration can occur using arbiters 24, 26.

[0021] While such a dual-arbiter architecture is useful, arbitration is separate and uncoordinated. Logic may be duplicated in arbiters 24, 26, wasting silicon area and perhaps adding to circuit propagation delays. With only 2 masters, only one "real" arbitration can occur at any time, either for the DRAM or for the SRAM, since typically a master cannot access both DRAM and SRAM at the same instant.

[0022] Figure 3 shows a single arbiter controlling access to separate memory devices in a 2-layer bus architecture. Dual-layer arbiter 30 receives memory-access requests from refresh controller 20 and from overlay engine 22. When the R_LCD request line from refresh controller 20 is activated, dual-layer arbiter 30 examines the SRAM-DRAM

(L_S/D) line which indicates whether refresh controller 20 desires to access SRAM 12 or DRAM 10. The L_S/D line can be a high-order address line or memory-select line that distinguishes between locations in DRAM 10 and in SRAM 12. For example, L_S/D high could select SRAM 12, while L_S/D low selects DRAM 10.

[0023] Likewise, when the R_VO request line from overlay engine 22 is activated, dual-layer arbiter 30 examines the SRAM-DRAM (V_S/D) line from overlay engine 22. V_S/D indicates whether overlay engine 22 desires to access SRAM 12 or DRAM 10.

[0024] In many cases, refresh controller 20 accesses SRAM 12 while overlay engine 22 accesses DRAM 10. Then dual-layer arbiter 30 allows simultaneous memory access. The grant line (GNT_LCD) to refresh controller 20 is activated to indicate that access to the requested memory has been granted to refresh controller 20. The select_A line to multiplexer (mux) A is set to cause mux 32 connect refresh controller 20 to SRAM 12. Then refresh controller 20 can access SRAM 12 over bus A through mux 32. The grant line (GNT_VO) to overlay engine 22 is set to indicate that overlay engine 22 has been granted access to DRAM 10 over bus B. SEL_B is driven low to allow mux 34 to connect

overlay engine 22 to bus B and DRAM 10.

[0025] When both requestors desire to access the same memory device, dual-layer arbiter 30 performs real arbitration. One of the requestors is denied access or delayed while the other requestor performs its memory access. A simple round-robin scheme could be used that alternates which requestor wins. For example, if refresh controller 20 won arbitration the last time, then overlay engine 22 is granted access the next time.

[0026] Round-robin arbitration may also be more random, such as by using a dual-phase clock. When both refresh controller 20 and overlay engine 22 make a simultaneous request during the first phase of the clock, then refresh controller 20 wins, but when the simultaneous request occurs in the second phase of the clock, then overlay engine 22 wins.

[0027] When one requestor has already gained access to the memory, then the later requestor must wait until the earlier requestor finishes accessing the memory. A limit can be placed on the size or length of the memory access.

[0028] For example, when refresh controller 20 activates its R_LCD request line and overlay engine 22 activates its R_VO1 request line at the same time, and both L_S/D and

V_S/D are high, dual-layer arbiter 30 chooses one or the other requestor. When refresh controller 20 is chosen, SEL_A is first driven high to allow overlay engine 22 to access SRAM 12 through mux 32. Once refresh controller 20 has completed access, SEL_A is driven low to allow overlay engine 22 to access SRAM 12 through mux 32. The control signals indicate that refresh controller 20 has access, then indicate that overlay engine 22 has access. A multi-bit grant line may be used that combines timing and selection information, or additional signals may be used.

[0029] Figure 4 shows a dual-layer arbiter with 3 requestors. Some graphics systems may have two video overlay engines. Dual-layer arbiter 40 receives requests from refresh controller 20, first overlay engine 22, and second overlay engine 23 on request lines R_LCD, R_VO1, R_VO2. Device-select lines L_S/D, V1_S/D, and V2_S/D are high when access to SRAM 12 is requested, but low when access to DRAM 10 is requested.

[0030] Dual-layer arbiter 30 arbitrates requests to two memory devices – SRAM 12 and DRAM 10. Each memory device has its own bus layer. Thus three requestors arbitrate for two memory devices in this embodiment.

[0031] Mux 42 can select either refresh controller 20, first over-

lay engine 22, or second overlay engine 23 to connect to bus A and SRAM 12. The SEL_A signal from dual-layer arbiter 40 can be a 2-bit signal to indicate which of 3 requestors is selected. Likewise, SEL_B from dual-layer arbiter 40 instructs mux 44 to select either refresh controller 20, first overlay engine 22, or second overlay engine 23 to be connected to bus B and DRAM 10.

[0032] Two-layer bus matrix 48 contains address, data, and control signals for bus A and bus B. Individual signals in the two buses are kept separate at any particular time, but routing area and other bus resources may be shared. A single arbitration state machine is used, making the two-layer bus matrix appear to be a single layer to the requestors.

[0033] Figure 5 details signals to and from the dual-layer arbiter with three requestors. Each requestor has a pair of request-grant lines that carry request-grant handshake signals. For example, refresh controller 20 activates its request signal REQ_LCD to signal to dual-layer arbiter 40 that it requests memory access. Device signal L_S/D is high, indicating that access to SRAM 12 is requested rather than to DRAM 10.

[0034] When refresh controller 20 wins arbitration, or when there

are no other requestors to DRAM 10, then dual-layer arbiter 40 activates grant signal GNT_LCD to let refresh controller 20 know that it has been granted access to SRAM 12. Dual-layer arbiter 40 drives SEL_A to indicate that mux 42 selects lines from refresh controller 20 to connect to bus A and SRAM 12.

[0035] Once mux 42 has connected refresh controller 20 to bus A, another set of handshake signals between dual-layer arbiter 40 and two-layer bus matrix 48 help perform the memory access. Dual-layer arbiter 40 activates the grant line to indicate that the A bus is ready to begin access. Two-layer bus matrix 48 responds with a ready signal RDY_A when SRAM 12 is ready to allow access.

[0036] Similar control signal SEL_B from dual-layer arbiter 40 controls mux 44 and two-layer bus matrix 48, which generates RDY_B as an acknowledgement back to dual-layer arbiter 40. First and second video overlay engines 22, 23 also generate request handshake signals REQ_VO1, REQ_VO2 and receive grant handshake signals GNT_VO1, GNT_VO2 from dual-layer arbiter 40.

[0037] When a new requestor is denied access or has to wait for an earlier requestor to finish access, dual-layer arbiter 40 does not immediately return the grant signal back to the

new requestor. The new requestor cannot begin access until its grant signal is activated.

[0038] Figure 6 shows a more sophisticated embodiment of a dual-layer arbiter that prioritizes the refresh controller. While a simple round-robin arbitration scheme is often preferred, a more complex scheme may also be used in some embodiments.

[0039] Arbitration logic for the two buses (bus A to SRAM, bus B to DRAM) can be shared, potentially reducing area, complexity, and cost. Device select and request signals are combined for each of the three requestors. AND gate 82 generates LC_A when the refresh controller requests access to the SRAM (A-bus) while AND gate 83 generates LC_B when the refresh controller requests access to the DRAM (B-bus).

[0040] Similarly, AND gate 84 generates V1_A when the first video overlay engine requests access to the SRAM (A-bus) while AND gate 85 generates V1_B when it requests access to the DRAM (B-bus). For the second video overlay engine, AND gate 86 generates V2_A when the request is to the SRAM (A-bus) while AND gate 87 generates V2_B when the request is to the DRAM (B-bus).

[0041] Flip-flop 81 acts as a toggle flip-flop, since its has its QB

output fed back to its D input. Output RR1 is a toggled signal that can implement a round-robin scheme, since RR1 alternates high and low with each clock or grant. Round-robin can be used for arbitrating between the first and second video overlay engines.

[0042] Arbiter state machine 90 receives pre-grant request inputs for each of the six possible requestor-memory combinations. State machine 90 then selects the highest priority pre-grant input and activates grant signals such as GNT_LCD, GNT_VO1, and GNT_VO2 to the requestors. State machine 90 can generate more complex timing signals, or can activate other state machines that control the exact timing of bus transfers and memory accesses.

[0043] AND gate 91 activates PG_LC_A to indicate that the refresh controller should win arbitration for the A-bus (SRAM) when neither the first or second video overlay engines request the A-bus. Likewise, AND gate 92 activates PG_LC_B to indicate that the refresh controller should win arbitration for the B-bus (DRAM) when neither the first or second video overlay engines request the B-bus.

[0044] OR-AND gate 93 activates PG_V1_A to indicate that the first video overlay engine should win arbitration for the SRAM when either the second video overlay engine does

not request the SRAM or the toggle signal RR1 favors the first video overlay engine over the second video overlay engine. OR-AND gate 94 generates PG_V1_B for the similar condition for the B-bus. OR-AND gates 95, 96 generate PG_V2_A, PG_V2_B for similar conditions for the second video overlay engine.

[0045] The conditions detected by the pre-grant request inputs are cases where real arbitration is not necessary, such as when requestors are requesting different memory resources. When two or more pre-grant request inputs are active, state machine 90 can grant access to both requestors when they are requesting different memory resources.

[0046] State machine 90 also receives the raw request lines LC_A, LC_B, V1_A, V1_B, V2_A, and V2_B. State machine 90 can perform real arbitration when two requestors are requesting the same memory, such as when LC_A and V1_A are both active. PG_V1_A could be active, showing that V1 has won the round-robin arbitration between V1 and V2. Then state machine 90 can arbitrate between the first video overlay engine and refresh controller. State machine 90 can choose the highest priority input, refresh controller, or it can use another layer of round-robin, alternately se-

lecting refresh controller and the overlay engines. Another toggle flip-flop could be used to implement round-robin arbitration with the refresh controller, or prioritizing logic can be included in state machine 90.

[0047] Figure 7 is a waveform illustrating arbitration using the dual-layer arbiter. The refresh controller keeps its request line REQ_LCD active (high). Initially the refresh controller has been granted access to the SRAM, and is performing a burst data access as its transaction TRANS_LCD.

[0048] However, at the 3rd clock pulse, a second requestor, the first video overlay engine, activates its request line REQ_VO1, with its V1_S/D line high (not shown) to indicate SRAM device selection.

[0049] The dual-layer arbiter grants the video overlay engine access, as a round-robin arbitration scheme allows access by other requestors, preventing the refresh controller from hogging the SRAM bus. The dual-layer arbiter kicks the refresh controller off the SRAM bus by de-activating the grant line GNT_LCD to the refresh controller. The burst access for the refresh controller ends.

[0050] The two-layer bus matrix de-activates RDY_A. The falling RDY_A is passed back to the refresh controller 20 as RDY_LCD.

[0051] When the dual-layer arbiter de-activates GNT_LCD, it also activates GNT_V1 to indicate that the first video refresh controller has won arbitration. The grant bus-A signal to the two-layer bus matrix 48 is again activated, and the two-layer bus matrix responds by activating RDY_A (not shown), which is passed back to the first video overlay engine as RDY_VO1 to indicate to the overlay engine that it may begin access. The first video overlay engine begins the active burst address and data transfers as bus transactions, shown as TRANS_VO1.

[0052] ALTERNATE EMBODIMENTS

[0053] Several other embodiments are contemplated by the inventor. A memory management unit or memory mapper external to refresh controller 20 and overlay engine 22 may be used to generate the DRAM-SRAM select lines L_S/D, V_S/D, or these lines may be generated by the masters themselves. Muxes may be bus switches or pass transistors that connect bit lines and control line on one bus to another bus. Buses A and B can differ in the number of address and data lines, and in the number and type of control lines. For example, SRAM 12 may be smaller than DRAM 10 and require fewer address bits. DRAM 10 may require different strobe control signals such as RAS

and CAS. Address and data lines can be separate or can share the same physical lines by being time-multiplexed. Other memory types such as FLASH or ROM types are possible variations.

[0054] An additional memory controller may be used for DRAM 10, such as to generate lower-level RAS and CAS control signals from higher-level request signals from refresh controller 20 or overlay engine 22. The exact timing and meaning of request, grant, and ready handshake signals can vary with different implementations and embodiments. Arbitration may be pipelined, masking some of the decisions. For example, one requestor's request may be delayed by pipelining, allowing a later request by a non-pipelined requestor to arrive at the dual-layer arbiter first.

[0055] Various bus protocols are possible. For example, the grant can be given to a particular requestor as an indication that the requestor will be the next requestor granted to the bus even when there is a currently-active bus transaction. The ready signal can be used to indicate exactly when the requestor should start accessing. Two separate grants GNT_LCD and GNT_V1 could be used, or a single grant could be used for a basic 2-layer arbiter.

[0056] An additional arbiter channel may be used for arbitrating

DRAM refresh cycles, or a hidden refresh scheme may be used. Additional requestors may be added to the arbitration, and may share a channel or have separate channels. Arbitration may be performed first among the additional requestors, then with the refresh controller and overlay engine. Display pixels may be further altered by the refresh controller, such as by color mapping, highlighting, inverting, clipping, etc. or for re-formatting for specific display types. The muxes can be bi-directional, allowing data to be returned from memory to the requestors during a READ, or data to flow in the other direction to the memories for a WRITE.

[0057] The ready signal can be generated by the memory (SRAM or DRAM) controller. The bus matrix can multiplex the two ready signals and pass the correct ready signal to the active requestor. The ready signal can have two meanings: 1- during a transfer, ready can be a cycle-by-cycle indicator as data is ready/valid; 2- during idle cycles, ready can indicate whether the DRAM or SRAM memory system is ready to accept new accesses or not from the granted requestor. There can be a case where a requestor obtains the grant from the arbiter while the memory controller is not ready to be accessed. Typically, the same ready signal

can be used for all 3 requestors in this case. Only the granted requestor needs to sample the ready signal. The two separate physical memories could actually be of the same type if a high-level of data access parallelism is required without the real need of using memories with different characteristics like latencies and costs.

[0058] The abstract of the disclosure is provided to comply with the rules requiring an abstract, which will allow a searcher to quickly ascertain the subject matter of the technical disclosure of any patent issued from this disclosure. It is submitted with the understanding that it will not be used to interpret or limit the scope or meaning of the claims. 37 C.F.R. § 1.72(b). Any advantages and benefits described may not apply to all embodiments of the invention. When the word "means" is recited in a claim element, Applicant intends for the claim element to fall under 35 USC § 112, paragraph 6. Often a label of one or more words precedes the word "means". The word or words preceding the word "means" is a label intended to ease referencing of claims elements and is not intended to convey a structural limitation. Such means-plus-function claims are intended to cover not only the structures described herein for performing the function and their

structural equivalents, but also equivalent structures. For example, although a nail and a screw have different structures, they are equivalent structures since they both perform the function of fastening. Claims that do not use the word means are not intended to fall under 35 USC § 112, paragraph 6. Signals are typically electronic signals, but may be optical signals such as can be carried over a fiber optic line.

[0059] The foregoing description of the embodiments of the invention has been presented for the purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form disclosed. Many modifications and variations are possible in light of the above teaching. It is intended that the scope of the invention be limited not by this detailed description, but rather by the claims appended hereto.